Quiz 5: 1/28/15

Question 1

List 3 things that must be saved in the process table during a context switch..

- Registers
- Link Register
- Program Counter
- Status Register
- Stack Pointer
- Floating Point Context
- Open File Descriptors
- Process State

Question 2

Write (pseudo) code to preform a context switch from Process A which is currently running to Process B. Assume that `struct process_table_entry` has whatever fields needed. Also assume you are in kernel mode. Just use pseudo code for any assembly instructions you need.

You only needed to fill in the `do_context_switch()` function. Below includes some other code to show better how this function might be implemented in a real OS.

```
struct process_table_entry
{
    register_t registers[NUM_REGISTERS];
    register_t stack_pointer;
    register_t status_register;
    enum process_state state;
};
```

```
struct process_table_entry processes[MAX_NUM_PROCESSES];
pid_t running_pid = 0; // PID 0 is a background process

void timer_interrupt_routine(void)
{
    // determin which process to run next
    pid_t next_process = run_scheduler(running_pid);

    if(next_process != running_pid)
    {
        do_context_switch(&processes[running_pid], &processes[next_process]);
        running_pid = 0;
    }
    return; // switch back to user mode with the new process running
}

void do_context_switch(struct process_table_entry* A,
                       struct process_table_entry* B)
{
    // save the context of A
    A->registers = user registers
    A->stack_pointer = user current stack pointer
    A->link_register = user current link register
    A->status_register = user current status register
    // restore the context of B
    user registers = B->registers
    stack pointer = B->stack_pointer
    link register = B->link_register
    status register = B->status_register
}
```